

Mean algorithms

Let a and b be positive numbers. The arithmetic, geometric, and harmonic means of a and b are, respectively,

$$A(a,b) = \frac{a+b}{2}, \quad G(a,b) = \sqrt{ab},$$

and

$$H(a,b) = \frac{2ab}{a+b} = \frac{2}{\frac{1}{a} + \frac{1}{b}} = \frac{1}{A(\frac{1}{a}, \frac{1}{b})}$$

Clearly,

$$A(a,a) = G(a,a) = H(a,a) = a.$$

Example. $a = 1, b = 2$.

$$A(1,2) = \frac{1+2}{2} = \frac{3}{2} = 1.5,$$

$$G(1,2) = \sqrt{1 \cdot 2} = \sqrt{2} = 1.41421\cdots,$$

$$H(1,2) = \frac{4}{1+2} = \frac{4}{3} = 1.33333\cdots.$$

Property. $0 < a < b \implies$

$$a < H(a,b) < G(a,b) < A(a,b) < b.$$

Proof. $0 < \left(\frac{a-b}{2}\right)^2 = \left(\frac{a+b}{2}\right)^2 - ab,$

that is $G^2(a,b) < A^2(a,b)$, or $G(a,b) < A(a,b)$.

Clearly, $A(a,b) < b$. The others follow.

from these, and the facts that
 $H(a, b) = 1/A(\frac{1}{a}, \frac{1}{b})$, $G(a, b) = 1/G(\frac{1}{a}, \frac{1}{b})$. ■

The arithmetic-geometric mean
algorithm of Gauss.

Given: $0 < a < b$,

$$a_0 = a, b_0 = b,$$

for $n = 0, 1, 2, 3, \dots$

$$\begin{cases} a_{n+1} = G(a_n, b_n), \\ b_{n+1} = A(a_n, b_n) \end{cases}$$

Example. $a = 1, b = \sqrt{2}$.

n	a_n	b_n
0	1.0000 00000	1.4142 13562
1	1.1892 07115	1.2071 06781
2	1.1981 23522	1.1981 56948
3	1.1981 40235	1.1981 40235

At the age of 14 Gauss carried out similar computations, by hand with 20 significant figures. Later by looking at the numerical results he established a connection between this algorithm and the theory of elliptic functions, and hence obtained fast algorithms for their computation.

We now give a partial analysis of the algorithm of Gauss. First of all,
 $0 < a = a_0 < a_1 < a_2 < \dots < b_1 < b_2 < b_0 = b$,

that is, in general

$$a_n < a_{n+1} < b_{n+1} < b_n.$$

We shall show that

$$b_n - a_n \rightarrow 0, n \rightarrow \infty.$$

It will follow that the common limit
 $\lim a_n = \lim b_n \equiv M(a, b)$,

exists. $M(a, b)$ is the Gauss arithmetic-geometric mean of a and b . Now,

$$\begin{aligned} b_{n+1} \pm a_{n+1} &= \frac{a_n + b_n}{2} \pm \sqrt{a_n b_n} \\ &= \frac{a_n \pm 2\sqrt{a_n b_n} + b_n}{2} \\ &= \frac{(\sqrt{b_n} \pm \sqrt{a_n})^2}{2}. \end{aligned}$$

Hence,

$$\begin{aligned} \frac{b_{n+1} - a_{n+1}}{b_{n+1} + a_{n+1}} &= \left(\frac{\sqrt{b_n} - \sqrt{a_n}}{\sqrt{b_n} + \sqrt{a_n}} \right)^2 \left(\frac{\sqrt{b_n} + \sqrt{a_n}}{\sqrt{b_n} + \sqrt{a_n}} \right)^2 \\ &= \left(\frac{b_n - a_n}{b_n + 2\sqrt{a_n b_n} + a_n} \right)^2 \end{aligned}$$

$$< \left(\frac{b_n - a_n}{b_n + a_n} \right)^2.$$

By induction on n ,

$$\frac{b_n - a_n}{b_n + a_n} < \left(\frac{b - a}{b + a} \right)^{2^n}$$

In other words, using $a_n < b_n < b$,

$$0 < b_n - a_n < 2b \left(\frac{b - a}{b + a} \right)^{2^n} \rightarrow 0, n \rightarrow \infty.$$

Gauss went on to show that, for $0 < k < 1$,

$$\frac{\pi}{2M(\sqrt{1-k^2}, 1)} = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1-k^2 \sin^2 \theta}} = \int_0^1 \frac{dt}{\sqrt{(1-t^2)(1-k^2 t^2)}}$$

This is a little advanced, so we'll give the analogous studies for some slightly different algorithms.

Problem. Show that $M(a, b) = a M(1, \frac{b}{a})$.

Hint: similar relations are true for all the means: $A(a, b) = a A(1, \frac{b}{a})$, $G(a, b) = a G(1, \frac{b}{a})$, $H(a, b) = a H(1, \frac{b}{a})$.

The arithmetic-harmonic mean algorithm.

Given: $0 < a < b$,

$$a_0 = a, b_0 = b,$$

-- for $n = 0, 1, 2, 3, \dots$

$$\begin{cases} a_{n+1} = H(a_n, b_n), \\ b_{n+1} = A(a_n, b_n). \end{cases}$$

Example. $a = 1, b = 2$

n	a_n	b_n
0	1.0000 00000	2.0000 00000
1	1.3333 33333	1.5000 00000
2	1.4117 64706	1.4166 66667
3	1.4142 11439	1.4142 15687
4	1.4142 13563	1.4142 13563

Clearly, in general,

$$0 < a = a_0 < a_1 < a_2 < \dots < b_2 < b_1 < b_0 = b.$$

We show that

$$b_n - a_n \rightarrow 0, n \rightarrow \infty.$$

First, we have

$$a_{n+1}, b_{n+1} = \frac{2a_n b_n}{a_n + b_n} = \frac{a_n + b_n}{2} = ab = \dots$$

identically in n . That is, $a_n b_n$ is an "invariant" for the pair of "difference equations"

$$a_{n+1} = H(a_n, b_n), \quad b_{n+1} = A(a_n, b_n).$$

Now,

$$\begin{aligned}
 \left(\frac{\sqrt{b_n} - \sqrt{a_n}}{\sqrt{b_n} + \sqrt{a_n}} \right)^2 &= \frac{b_n - 2\sqrt{a_n b_n} + a_n}{b_n + 2\sqrt{a_n b_n} + a_n} \\
 &= \frac{\frac{a_n + b_n}{2} - \sqrt{a_n b_n}}{\frac{a_n + b_n}{2} + \sqrt{a_n b_n}} \\
 &= \frac{b_{n+1} - \sqrt{a_{n+1} b_{n+1}}}{b_{n+1} + \sqrt{a_{n+1} b_{n+1}}} \\
 &= \frac{\sqrt{b_{n+1}} - \sqrt{a_{n+1}}}{\sqrt{b_{n+1}} + \sqrt{a_{n+1}}}.
 \end{aligned}$$

By induction,

$$\frac{\sqrt{b_n} - \sqrt{a_n}}{\sqrt{b_n} + \sqrt{a_n}} = \left(\frac{\sqrt{b} - \sqrt{a}}{\sqrt{b} + \sqrt{a}} \right)^{2^n} \rightarrow 0, n \rightarrow \infty.$$

That is,

$$\begin{aligned}
 0 < b_n - a_n &= (\sqrt{b_n} + \sqrt{a_n})(\sqrt{b_n} - \sqrt{a_n}) \\
 &= (\sqrt{b_n} + \sqrt{a_n})^2 \left(\frac{\sqrt{b} - \sqrt{a}}{\sqrt{b} + \sqrt{a}} \right)^{2^n} \\
 &< 4b \left(\frac{\sqrt{b} - \sqrt{a}}{\sqrt{b} + \sqrt{a}} \right)^{2^n} \rightarrow 0, n \rightarrow \infty
 \end{aligned}$$

We have thus shown that

$$a_n \rightarrow d, b_n \rightarrow d, n \rightarrow \infty,$$

and, indeed, quadratically fast. Now using the invariance, it is easy to evaluate the limit d . For we have

$$a_n b_n = ab = c.$$

Hence, letting $n \rightarrow \infty$, we get

$$d = \lim a_n = \lim b_n = \sqrt{ab} = G(a, b).$$

That is, d , the arithmetic-harmonic mean of a and b , is just the geometric mean of a and b !

Heron's algorithm for \sqrt{c} is gotten by eliminating a_n from the above process, using $a_n b_n = c$:

$$b_{n+1} = \frac{a_n + b_n}{2} = \frac{1}{2} \left(b_n + \frac{c}{b_n} \right)$$

Problem. Show that

$$b_n = \sqrt{c} \frac{1 + \rho^{2^n}}{1 - \rho^{2^n}} : \rho = \frac{b_0 - \sqrt{c}}{b_0 + \sqrt{c}}$$

and hence that Heron's algorithm converges quadratically to \sqrt{c} whenever $b_0 > 0$. Hint: you need $|\rho| < 1$ for this.

We could just as well have eliminated b_n to get the algorithm

$$a_{n+1} = \frac{2a_n b_n}{a_n + b_n} = \frac{2c a_n}{a_n^2 + c}$$

Problem. Show that

$$a_n = \sqrt{c} \frac{1 + \sigma^{2^n}}{1 - \sigma^{2^n}} : \sigma = \frac{\sqrt{c} - a_0}{\sqrt{c} + a_0}$$

(8)

and hence that $a_n \rightarrow \sqrt{c}$ quadratically whenever $a_0 > 0$.

Remark. If we drop a_n (or b_n) from the algorithm we lose the nice feature of "bracketing": $a_n < \sqrt{c} < b_n$.

Newton's method for $f(x) = 0$ starts with an approximation, x_0 , to x , uses the Taylor development to linearize the equation about the point x_0 :

$$0 = f(x) = f(x_0 + (x - x_0)) \\ \quad \quad \quad = f(x_0) + f'(x_0)(x - x_0),$$

solves the resulting linear equation, calling the solution x , and then iterates the process:

$$x_0 \doteq x,$$

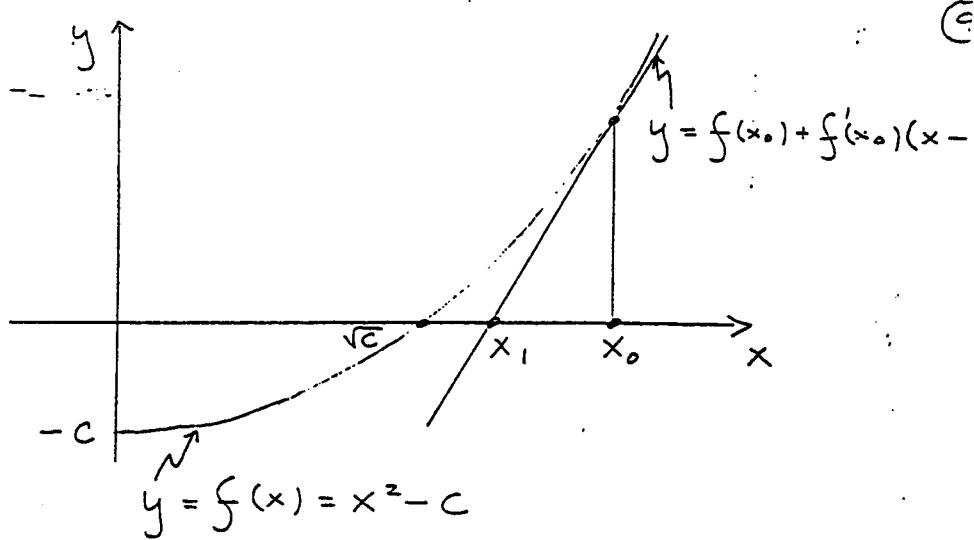
for $n = 0, 1, 2, 3, \dots$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

For $f(x) = x^2 - c$, we have $f'(x) = 2x$, so

$$x_{n+1} = x_n - \frac{x_n^2 - c}{2x_n} = \frac{1}{2} \left(x_n + \frac{c}{x_n} \right),$$

which is just Heron's algorithm.



Problem. Express the iteration

$$x_{n+1} = \frac{2cx_n}{x_n^2 + c}$$

as a Newton iteration for solving $g(x) = 0$, where $g(\sqrt{c}) = 0$. That is, find all functions $g(x)$ so that

$$x_{n+1} = x_n - \frac{g(x_n)}{g'(x_n)}$$

for every choice of x_0 . Hint:

$$\frac{g'(x)}{g(x)} = \frac{d}{dx} \log g(x) = \frac{1}{x} \frac{x^2 + c}{x^2 - c};$$

use partial fractions. Answer:

$$g(x) = x \frac{x^2 - c}{x} : x \neq 0 \text{ arbitrary.}$$

(16)

Problem Show that Newton's iteration for $f(x) = \frac{1}{x^2} - \frac{1}{c} = 0$ is

$$x_{n+1} = x_n + \frac{x_n}{2} \left(1 - \frac{x_n^2}{c} \right).$$

Compute $\sqrt{2}$ to 10 significant figures, starting with $x_0 = 1$.

Problem (Reciprocals without division).

Show that Newton's iteration for $f(x) = \frac{1}{x} - c = 0$ is

$$x_{n+1} = x_n (2 - cx_n).$$

Also show that

$$1 - cx_n = (1 - cx_0)^{2^n},$$

and hence that $x_n \rightarrow \frac{1}{c}$ if x_0 is such that $0 < cx_0 < 2$.

Example $c = 3, x_0 = 0.3$

n	x_n
0	0.30000 00000 00000 0
1	0.33000 00000 00000 0
2	0.33330 00000 00000 0
3	0.33333 33300 00000 0
4	0.33333 33333 33333 3

The number of correct digits doubles at each iteration.

Computing π , with Archimedes.

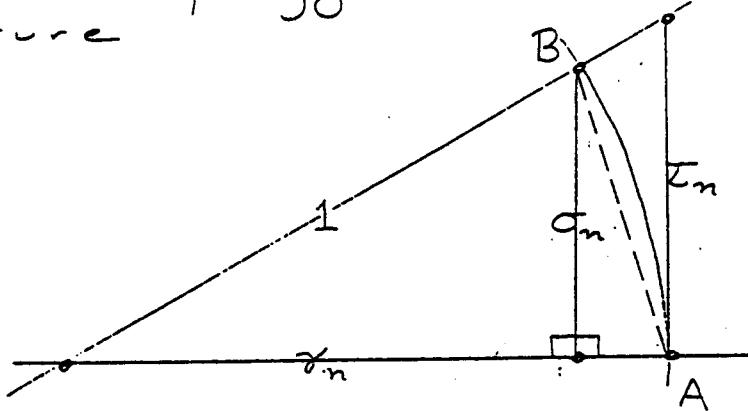
Let $s_n(t_n)$ be one half of the perimeter of a regular polygon with 2^n sides inscribed in (circumscribed about) a circle of radius 1. We have

$$s_1 = 2, \quad t_1 = +\infty,$$

as (degenerate) initial conditions
Also,

$$s_n = 2^n \sigma_n, \quad t_n = 2^n \tau_n,$$

where $\sigma_n(\tau_n)$ is half the length of side of the inscribed (circumscribed) polygon. From the picture



we deduce that

$$\frac{\tau_n}{1} = \frac{\sigma_n}{\tau_n} \quad (\text{similar triangles})$$

$$1 = \tau_n^2 + \sigma_n^2 \quad (\text{Pythagoras})$$

(12)

and that, again by Pythagoras,

$$\begin{aligned}\overline{AB}^2 &= (2\sigma_{n+1})^2 = \sigma_n^2 + (1-\gamma_n)^2 \\ &= \sigma_n^2 + 1 - 2\gamma_n + \gamma_n^2 \\ &= 2(1-\gamma_n).\end{aligned}$$

Consequently,

$$\sigma_{n+1}^2 = \frac{1}{2}(1-\gamma_n),$$

$$\gamma_{n+1}^2 = 1 - \sigma_{n+1}^2 = \frac{1}{2}(1+\gamma_n),$$

$$\begin{aligned}\tau_{n+1} &= \frac{\sigma_{n+1}}{\gamma_{n+1}} = \left(\frac{1-\gamma_n}{1+\gamma_n}\right)^{1/2} \left(\frac{1+\gamma_n}{1+\gamma_n}\right)^{1/2} \\ &= \frac{\sigma_n}{1+\gamma_n} = \frac{\sigma_n}{1+\frac{\sigma_n}{\tau_n}} = \frac{\sigma_n \tau_n}{\sigma_n + \tau_n}, \\ \sigma_{n+1} &= \left(\frac{1-\gamma_n}{2}\right)^{1/2} \left(\frac{1+\gamma_n}{1+\gamma_n}\right)^{1/2} \\ &= \frac{1}{2} \frac{\sigma_n}{\left(\frac{1+\gamma_n}{2}\right)^{1/2}} = \frac{\sigma_n}{2\gamma_{n+1}} = \frac{\sigma_n \tau_{n+1}}{2\sigma_{n+1}}, \\ &= \left(\frac{\sigma_n \tau_{n+1}}{2}\right)^{1/2}.\end{aligned}$$

It follows that

$$t_1 = +\infty, s_1 = 2,$$

for $n = 1, 2, 3, \dots$

$$\begin{cases} t_{n+1} = H(s_n, t_n), \\ s_{n+1} = G(s_n, t_{n+1}) \end{cases}$$

This is a harmonic-geometric mean algorithm of Borchardt type, with special initial conditions. ("Borchardt type" refers to the use of t_{n+1} at the second step, instead of t_n .) Let's run it!

Example.

n	s_n	t_n
0	2.0000 00000	+∞
1	2.8284 27125	4.0000 00000
2	3.0614 67459	3.3137 08498
3	3.1214 45152	3.1825 97878
4	3.1365 48491	3.1517 24908
5	3.1403 31158	3.1441 18386
6	3.1412 77252	3.1422 23630
7	3.1415 13802	3.1417 50370
8	3.1415 72941	3.1416 32082
9	3.1415 87726	3.1416 02512
10	3.1415 91422	3.1415 95118
11	3.1415 92346	3.1415 93270
12	3.1415 92577	3.1415 92808
13	3.1415 92634	3.1415 92692
14	3.1415 92648	3.1415 92662
15	3.1415 92651	3.1415 92654
16	3.1415 92652	3.1415 92652

Thus, Archimedes needed $2^{16} = 65,536$ sides to compute π accurately to

than significant figures. It is curious that this algorithm is so much slower than those of "Gauss type".

Let us now use our knowledge of trigonometry. We have

$$s_n = 2^n \sin \frac{\pi}{2^n} \rightarrow \pi, n \rightarrow \infty$$

$$t_n = 2^n \tan \frac{\pi}{2^n} \rightarrow \pi, n \rightarrow \infty.$$

At the same time, we can choose another angle θ , rather than π , and drop the initial index back by 1. Then,

$$s_n = 2^n \sin \frac{\theta}{2^n} \rightarrow \theta, n \rightarrow \infty$$

$$t_n = 2^n \tan \frac{\theta}{2^n} \rightarrow \theta, n \rightarrow \infty,$$

and

$$s_0 = \sin \theta, t_0 = \tan \theta.$$

The above derivation is really using the $1/2$ -angle formulas

$$\sin \frac{\theta}{2} = \left(\frac{1 - \cos \theta}{2} \right)^{1/2}, \cos \frac{\theta}{2} = \left(\frac{1 + \cos \theta}{2} \right)^{1/2}.$$

In order that these be valid, an

$0 \leq \sin \theta \leq \tan \theta$, we must insist that $0 \leq \theta \leq \frac{\pi}{2}$. The algorithm becomes

$$s_0 = \sin \theta, t_0 = \tan \theta,$$

for $n = 0, 1, 2, 3, \dots$

$$\begin{cases} t_{n+1} = H(s_n, t_n), \\ s_{n+1} = G(s_n, t_{n+1}). \end{cases}$$

Finally, let us put

$$x = \sin \theta, \theta = \arcsin x,$$

so that

$$\cos \theta = \sqrt{1-x^2}, \tan \theta = \frac{x}{\sqrt{1-x^2}}.$$

Then we get the algorithm

$$s_0 = x, t_0 = x/\sqrt{1-x^2}$$

for $n = 0, 1, 2, 3, \dots$

$$\begin{cases} t_{n+1} = H(s_n, t_n), \\ s_{n+1} = G(s_n, t_{n+1}). \end{cases}$$

Here, we must restrict

$$0 \leq x \leq 1,$$

and we then have

$$x = s_0 < s_1 < s_2 < \dots < t_2 < t_1 < t_0 = x/\sqrt{1-x^2}$$

and

$$\lim s_n = \lim t_n = \arcsin x.$$

Moreover, from

$$\sin z = z - \frac{z^3}{3!} + \frac{z^5}{5!} - \dots, |z| < \infty.$$

$$\tan z = z + \frac{z^3}{3} + \frac{2z^5}{15} + \dots, |z| < \frac{\pi}{2},$$

we find that

$$s_n = \theta - \frac{\theta^2}{6 \cdot 4^n} + O\left(\frac{1}{16^n}\right), n \rightarrow \infty,$$

$$t_n = \theta + \frac{\theta^2}{3 \cdot 4^n} + O\left(\frac{1}{16^n}\right), n \rightarrow \infty,$$

where $\theta = \arcsin x$.

Problem. Show that, for $x \geq 0$, if

$$t_0 = x/\sqrt{1+x^2}, \quad s_0 = x,$$

then

$$x/\sqrt{1+x^2} = t_0 < t_1 < t_2 < \dots < s_2 < s_1 < s_0 = x,$$

and

$$\lim s_n = \lim t_n = \operatorname{arsinh} x$$

$$\text{Hint: } t_n = 2^n \tanh \frac{\theta}{2^n}, \quad s_n = 2^n \sinh \frac{\theta}{2^n},$$

$$\sinh \frac{\theta}{2} = \left(\frac{\cosh \theta - 1}{2} \right)^{1/2}, \quad \cosh \frac{\theta}{2} = \left(\frac{\cosh \theta + 1}{2} \right)^{1/2}.$$

Look up other facts, as needed.

Problem. Compute $\operatorname{arcsinh} i$ to (about) 15 significant figures.
Check, using $\operatorname{arcsinh} x = \log(x + \sqrt{1+x^2})$

Problem. Apply the arithmetic-geometric mean algorithm of Borchardt type

$$a_0 = a, b_0 = b,$$

$$\text{for } n = 0, 1, 2, 3, \dots$$

$$b_{n+1} = A(a_n, b_n),$$

$$a_{n+1} = G(a_n, b_{n+1}),$$

to compute the limit when $a=1, b=\sqrt{5}$. What is this limit? Relate the iterates a_n, b_n to the iterates s_n, t_n of the (generalized) Archimedean algorithm.

Problem (continuation). Show that

$$b_{n+1}^2 - a_{n+1}^2 = \frac{b_n^2 - a_n^2}{4},$$

and hence that

$$b_n^2 - a_n^2 = \frac{b^2 - a^2}{4^n}.$$

(Thus, $4^n(b_n^2 - a_n^2)$ is the invariant.)

Problem: Let

$$R(a, b) = \int_0^{\pi/2} \frac{d\theta}{\sqrt{a^2 \cos^2 \theta + b^2 \sin^2 \theta}}$$

Show that

$$R(a, b) = \frac{1}{2} \int_0^\infty \frac{dt}{\sqrt{t(t+a^2)(t+b^2)}}$$

Hint: Let $\sin \theta = a/\sqrt{t+a^2}$.

Problem (continuation). Show that

$$R(a, b) = R(G(a, b), A(a, b)).$$

(Hence, $R(a, b)$ is the invariant for Gauss' algorithm.) Hint: Let
 $a_i = G(a, b)$, $b_i = A(a, b)$,

and

$$t = \frac{x(x+a_i^2)}{x+b_i^2} = \frac{x(x+ab)}{x+b^2}$$

Then,

$$\begin{aligned}\frac{dt}{dx} &= \frac{x^2 + 2b_i^2 x + a_i^2 b_i^2}{(x+b_i^2)^2} \\ &= \frac{x^2 + b_i(a+b)x + b_i^2 ab}{(x+b_i^2)^2} \\ &= \frac{(x+ab_i)(x+b_i b)}{(x+b_i^2)^2}.\end{aligned}$$

Problem (continuation). Show that

$$\frac{\pi}{2b_n} < R(a, b) < \frac{\pi}{2a_n},$$

and hence that

$$\frac{\pi}{2M(a, b)} = R(a, b) = \int_0^{\pi/2} \frac{d\theta}{\sqrt{a^2 \cos^2 \theta + b^2 \sin^2 \theta}}.$$

Hint. We have $R(a, b) = R(a_n, b_n) = \int_0^{\pi/2} \frac{d\theta}{\sqrt{\Delta_n(\theta)}}$, with $\Delta_n(\theta) = a_n^2 \cos^2 \theta + b_n^2 \sin^2 \theta = b_n^2 - (b_n^2 - a_n^2) \sin^2 \theta$. As θ increases from 0 to $\pi/2$, $\Delta_n(\theta)$ decreases from b_n^2 to a_n^2 .

Problem. Apply the compounded mean algorithm

$$a_0 = a, \quad b_0 = b$$

for $n = 0, 1, 2, 3, \dots$

$$\begin{cases} a_{n+1} = M(a_n, b_n), \\ b_{n+1} = A(a_n, b_n), \end{cases}$$

to compute the arithmetic-arithmetic geometric mean of $a=1, b=2$, to 10 or 15 significant figures. Does the algorithm appear to be quadratically convergent? What is the limit, as a function of a and b ?

(20)

Problem. Show that

$$\frac{\pi}{2M(1,\sqrt{2})} = \int_0^1 \frac{dt}{\sqrt{1-t^4}}.$$

Aitken's Δ^2 -process

We now try to speed up Archimedes' algorithm, and others.

Suppose we have a slowly convergent sequence, which we "guess" to be of the form

$$S_n \sim S^* + \alpha \lambda^n.$$

There are two cases, depending on whether λ is unknown or known. We consider the former first.

From the three values

$$S_n \sim S^* + \alpha \lambda^n,$$

$$S_{n+1} \sim S^* + \alpha \lambda^{n+1},$$

$$S_{n+2} \sim S^* + \alpha \lambda^{n+2},$$

we first deduce that

$$\Delta S_n \equiv S_{n+1} - S_n \sim \alpha(\lambda-1)\lambda^n$$

$$\Delta S_{n+1} = S_{n+2} - S_{n+1} \sim \alpha(\lambda-1)\lambda^{n+1}$$

and then that

$$\begin{aligned}\Delta^2 S_n &\equiv \Delta S_{n+1} - \Delta S_n \\ &= S_{n+2} - 2S_{n+1} + S_n \\ &\sim \alpha(\lambda-1)^2 \lambda^n.\end{aligned}$$

But then, assuming $\alpha \neq 0, \lambda \neq 1$.

(2)

$$\frac{(\Delta s_n)^2}{\Delta^2 s_n} \sim \frac{\alpha^2 (\lambda-1)^2 \lambda^{2n}}{\alpha (\lambda-1)^2 \lambda^n} = \alpha \lambda^n,$$

and so

$$s'_n = s_n - \frac{(\Delta s_n)^2}{\Delta^2 s_n} \sim s^*.$$

We have actually viewed the above three relations involving s_n, s_{n+1}, s_{n+2} as three (nonlinear) equations in the unknowns s^* , α , and λ . Then, we eliminated α and λ , and solved for s^* . We expect s'_n to converge "faster" than s_n .

Example Archimedes could have computed the better values s'_n :

n	s_n	Δs_n
0	2.0000 00000	0.8284 27125
1	2.8284 27125	0.2330 40334
2	3.0614 67459	0.0599 77693
3	3.1214 45152	0.0151 03339
4	3.1365 48491	0.0037 82667
5	3.1403 31158	0.0009 46094
6	3.1412 77252	0.0002 36550
7	3.1415 13802	0.0000 59139
8	3.1415 72941	0.0000 14785
9	3.1415 87726	

n	$\Delta^2 s_n$	s'_n
0	-0.595386791	3.152681772
1	-0.173062641	3.142231404
2	-0.044874354	3.141631814
3	-0.011320672	3.141595091
4	-0.002836573	3.141592807
5	-0.000709544	3.141592664
6	-0.000177411	3.141592655
7	-0.000044354	3.141592654

Thus, Archimedes, with the help of Aitken, really only needed $2^7 = 128$ sides!

Problem. Apply the Δ^2 -process to Archimedes' n -sequence. And to the sequences produced by the first two problems of mean algorithms - page 17.

Problem. Apply the Δ^2 -process, analytically, to the partial sums, $s_n(z) = 1 + z + z^2 + \dots + z^n$, of the geometric series $\frac{1}{1-z} = 1 + z + z^2 + z^3 + \dots$. This series converges for $|z| < 1$, and diverges for $|z| > 1$. How does Aitken do in the latter case?!!

Problem. If $s_n = a + t_n$, then $s'_n = a + t'_n$, where s'_n, t'_n are Aitken's derived sequences.

Problem. Compute s'_n explicitly for $s_n = s^* + \alpha \lambda^n + \beta \mu^n$.

Problem. Apply the Δ^2 -process to the Archimedes-Aitken sequence s'_n to get a new sequence s''_n . And then to get $s'''_n, s^{(+)}_n, \dots$. How many sides to get 10 significant figures of π ? Or 15?

The Padé table of a formal power series

$$f(z) = c_0 + c_1 z + c_2 z^2 + \dots$$

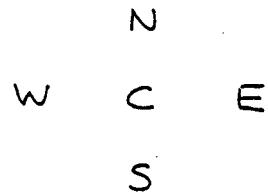
is an array of rational functions

	0	0	0	0	...
∞	$r_{0,0}(z)$	$r_{0,1}(z)$	$r_{0,2}(z)$	$r_{0,3}(z)$...
∞	$r_{1,0}(z)$	$r_{1,1}(z)$	$r_{1,2}(z)$	$r_{1,3}(z)$...
∞	$r_{2,0}(z)$	$r_{2,1}(z)$	$r_{2,2}(z)$	$r_{2,3}(z)$...
∞	$r_{3,0}(z)$	$r_{3,1}(z)$	$r_{3,2}(z)$	$r_{3,3}(z)$...
:	:	:	:	:	.

in which

$$-- r_{n,0}(z) = s_n(z) \equiv \sum_0^n c_k z^k, \quad (5)$$

and on a typical "star",



in the table, we compute E from

$$\frac{1}{N-C} + \frac{1}{S-C} = \frac{1}{W-C} + \frac{1}{E-C}.$$

(The algorithm may "fail" in "abnormal" cases.)

Problem. Express $r_{n,1}(1)$ in terms of $s_n = s_n(1)$. Hence, deduce that $r_{n,1}(1) = s'_n$ - Aitken's sequence.

Problem. Find $r_{n,v}(z)$, $0 \leq n \leq 1$, $0 \leq v \leq 1$, for $f(z) = e^z = 1 + z + \frac{z^2}{2!} + \frac{z^3}{3!} + \dots$

Problem. Find $r_{n,z}(1)$ for $s_n = s^* + \alpha \lambda^n + \beta \mu^n$.

Problem. Is $r_{n,z}(1) = s''_n$? (Padé acceleration is the correct generalization of the Δ^2 -process

to speed convergence of sequences
 $s_n = s^* + \sum_{r=1}^n \alpha_r \lambda_r^n$. We have $r_{n,r}(1)$
 $= s^*$ for $n \geq r$.)

Problem. How much could Padé have helped Archimedes? Compute $r_{n,n}(1)$ for $n = 0, 1, 2, 3, \dots$ and the Archimedean sequence $s_n = s_n(1)$. Compare with previous results.

The epsilon algorithm.

This is a more convenient way to organize Padé acceleration (and to solve the previous problem).

One computes the array

$$\begin{aligned} S_0 &= \varepsilon_1^{(0)} \\ S_1 &= \varepsilon_1^{(1)} \rightarrow \varepsilon_2^{(0)} \\ S_2 &= \varepsilon_1^{(2)} \rightarrow \varepsilon_2^{(1)} \rightarrow \varepsilon_3^{(0)} \rightarrow \varepsilon_4^{(0)} \\ S_3 &= \varepsilon_1^{(3)} \rightarrow \varepsilon_2^{(2)} \rightarrow \varepsilon_3^{(1)} \rightarrow \varepsilon_4^{(1)} \rightarrow \varepsilon_5^{(0)} \rightarrow \varepsilon_6^{(0)} \\ S_4 &= \varepsilon_1^{(4)} \rightarrow \varepsilon_2^{(3)} \rightarrow \varepsilon_3^{(2)} \rightarrow \varepsilon_4^{(2)} \rightarrow \varepsilon_5^{(1)} \rightarrow \varepsilon_6^{(1)} \rightarrow \varepsilon_7^{(0)} \\ S_5 &= \varepsilon_1^{(5)} \quad \vdots \quad \vdots \quad \vdots \quad \vdots \end{aligned}$$

(when possible) from the boundary values

$$\varepsilon_0^{(m)} = 0, \quad \varepsilon_1^{(m)} = s_m,$$

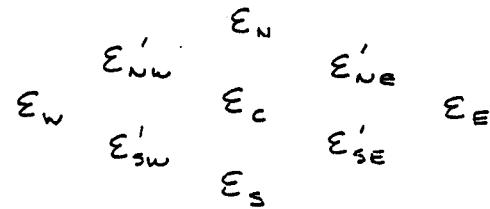
and the rhombus rules

$$(\varepsilon_{m+1}^{(m)} - \varepsilon_{m-1}^{(m)}) (\varepsilon_m^{(m+1)} - \varepsilon_m^{(m)}) = 1.$$

We then have

$$\varepsilon_{2m+1}^{(m)} = r_{m+m, m}(1).$$

Proof. Consider the "constellation"



From the rhombus rules

$$\begin{aligned} \frac{1}{\varepsilon_N - \varepsilon_c} + \frac{1}{\varepsilon_s - \varepsilon_c} &= (\varepsilon'_{NW} - \varepsilon'_{NE}) + (\varepsilon'_{SE} - \varepsilon'_{SW}) \\ &= (\varepsilon'_{NW} - \varepsilon'_{SW}) + (\varepsilon'_{SE} - \varepsilon'_{NE}) \\ &= \frac{1}{\varepsilon_w - \varepsilon_c} + \frac{1}{\varepsilon_E - \varepsilon_c}. \end{aligned}$$

Hence, by induction, if

$$\varepsilon_w = r_w(1), \quad \varepsilon_N = r_N(1), \quad \varepsilon_s = r_s(1),$$

then

$$\varepsilon_E = r_E(1).$$

Coding the epsilon algorithm.

At the completion of stage n we have the values

$$\varepsilon_1^{(n)}, \varepsilon_2^{(n-1)}, \varepsilon_3^{(n-2)}, \dots, \varepsilon_{n+1}^{(0)}$$

stored in a one dimensional array as

$$\varepsilon_n, \varepsilon_{n-1}, \varepsilon_{n-2}, \dots, \varepsilon_0.$$

We then introduce

$$s_{n+1} = \varepsilon_1^{(n+1)} = \varepsilon_{n+1},$$

and use the rhombus rules, and two additional temporary storage locations, to update our array. We have the algorithm

for $n \leftarrow 0, 1, 2, 3, \dots$

$$\tau \leftarrow 0, \varepsilon_n \leftarrow s_n,$$

for $m \leftarrow n-1, n-2, \dots, 1, 0$

$$\sigma \leftarrow \varepsilon_m,$$

$$\varepsilon_m \leftarrow \tau + 1 / (\varepsilon_{m+1} - \varepsilon_m)$$

$$\tau \leftarrow \sigma$$

if n is even then write ε_0 .

This outputs the sequence

$$r_{0,0}(1), r_{1,1}(1), r_{2,2}(1), r_{3,3}(1), \dots$$